

## RESEARCH

## Open Access

# A data dissemination mechanism for motorway environment in VANETs

Yang Yang<sup>1\*</sup>, Qian Liu<sup>1</sup>, Zhipeng Gao<sup>1</sup>, Xuesong Qiu<sup>1</sup>, Lanlan Rui<sup>1</sup> and Xin Li<sup>2</sup>

## Abstract

An efficient data dissemination mechanism is very critical for traffic safety and inquiring path information. All vehicles go straight and do not consider the effect of traffic lights during its running in the motorway. But in view of characteristic of motorway, the fast moving of vehicles will cause the frequent changes of network topology, which is a challenge for data dissemination for motorway environment in Vehicular *ad hoc* networks (VANETs). This paper proposes a novel data dissemination mechanism for VANETs which apply to broadcasting and on-demand unicast scenarios. We firstly propose a weighted  $K$ -nearest neighbors prediction algorithm with self-feedback to predict the current speed of a vehicle and construct a set of forwarders for emergency message broadcasting. To satisfy users' personal demands, the paper also designs a unicast mechanism based on information urgent level and the improved Greedy Perimeter Stateless Routing (GPSR) protocol. The simulation results show that our proposed method can achieve better performance with higher delivery ratio, lower number of forwarders, and less copies of messages.

**Keywords:** VANETs; Data dissemination; On demand; Delivery ratio

## 1 Introduction

Vehicular *ad hoc* networks (VANETs) are a special class of mobile *ad hoc* networks (MANETs) where nodes self-organize and self-manage in a distributed way. The networks consist of vehicles and/or roadside infrastructures that communicate with each other. Apart from conventional *ad hoc* networks, VANETs possess some special characteristics, such as highly dynamic topology and frequent disconnection [1,2].

In recent years, increasing research efforts have been put into VANETs, such as routing protocol, data dissemination, information security, etc. Among these research fields, the data dissemination mechanism has gained more attention as it determines whether users acquire on-demand services timely. Data dissemination in VANETs is frequently used as a way to disseminate messages (i.e., emergency accident, traffic information services, and advertisement) among a group of vehicles. Vehicles are equipped with the radio devices which allow the vehicles to communicate with others located within one-hop communication range.

Then, the messages will be disseminated and shared among a group of moving vehicles [3].

In the motorway environment, there are two types of messages routed on the network layer in VANETs currently:

- *Cooperative Awareness Messages (CAM)*. They contain basic status information like the current position, speed, acceleration, as well as the vehicle identifier. These parameter values are periodically exchanged by *hello* message between vehicles in one hop.
- *Decentralized Environmental Notification Messages (DENM)*. These messages may be related to traffic condition information, query/response message, emergency warning information, etc. It always uses the multi-hop mechanisms to efficiently deliver these messages.

Generally, the works of data dissemination in VANETs are mainly focused on *DENM* messages. Consequently, for emergency warning information, they are disseminated using broadcasting scheme. The main advantage of broadcasting scheme is that a vehicle does not need to know a

\* Correspondence: [yyang@bupt.edu.cn](mailto:yyang@bupt.edu.cn)

<sup>1</sup>State Key Laboratory of Networking and Switching Technology, Beijing University of Posts and Telecommunications, No. 10 Xi Tu Cheng Road, Beijing 100876, China

Full list of author information is available at the end of the article

route to a specific destination, and it spreads the emergency information rapidly. However, a large scale of broadcasting the same message will cause the problems of channel contention and packet collisions. Rebroadcasting due to transmission failure makes the problem even worse. This is called broadcast storm problem. To alleviate this problem, a common solution employed by most researchers is to reduce the number of redundant rebroadcast messages by only selecting part of vehicles for relaying messages. While as to simple user, he/she may have end-to-end communication demands. So, unicast is a more appropriate way for information inquiry demands. But some mechanisms have lower data delivery ratio when only considering the minimum path or deadline priority.

As vehicles move at high speed in the motorway, the topology formed by vehicles is always changing. The highly dynamic topology results in a frequent disconnection of link between two vehicles. Based on this, the *DENM* messages are classified into two types in the paper: the emergency message and the inquiry message. This paper proposes a data Dissemination Service for Broadcasting and on-Demand unicast in the VANETs (DSV-BD). We use different transmission methods to deliver different types of messages. For emergency messages, a proposed broadcasting algorithm can make the emergency messages cover a limited network domain rapidly. On the other hand, if some vehicles request real-time messages of personal interest, an on-demand unicast mechanism is proposed to reduce network load.

The rest of the paper is organized as follows. In Section 2, we investigate previous related works on data dissemination protocols in VANETs. The broadcasting algorithm for emergency message and the on-demand unicast mechanism for inquiry message are proposed in Section 3 and Section 4. In these two parts, we give the assumption of our system and introduce several terminologies and key mechanisms. Section 5 validates the solution by simulations. Finally, we give the conclusions of this paper in Section 6.

## 2 Related work

Flooding is a fundamental mechanism to implement the data dissemination in VANETs. Unfortunately, flooding introduces significant communication overhead due to redundant rebroadcasting. This will cause the well-known problem called broadcast storm problem. A common solution is to reduce the number of redundant rebroadcast packets. This is typically solved by selecting some special vehicles to relay the packet rather than rebroadcast it.

S.Y. Ni et al. [4] research five different classes of solutions: probabilistic schemes, counter-based schemes, distance-based schemes, location-based schemes, and cluster-based schemes for broadcast storm problem. Simulation results show that a counter-based scheme can reduce some redundant rebroadcasts when the host

distribution is dense. While location-based scheme can achieve higher performance when GPS service is available. S. Panichapiboon et al. [5] gave a review of variety data dissemination protocols for VANETs. The authors introduce a new metric called dissemination efficiency to discuss the performance evaluation of those protocols.

The urban multi-hop broadcast (UMB) protocol [6] is designed to solve the broadcast storm and the reliability in multi-hop broadcasting. The forward vehicle tries to select a far relay vehicle in the intended direction on a linear road through request to broadcast (RTB) packet and clear to broadcast (CTB) packet. In order to disseminate the message in all directions, a repeater vehicle is employed at the intersection to rebroadcast the message. UMB is inefficient in a sense that the next rebroadcast vehicle has to wait the longest before being able to transmit the CTB packet. This is because the longest black-burst duration is assigned to the next rebroadcast vehicle. G. Korkmaz et al. [7] proposed a fully *ad hoc* multi-hop broadcast protocol (AMB) for inter-vehicular networks. The AMB is an *ad hoc* extension of the UMB protocol. The protocol eliminates drawbacks of the UMB protocol by employing an efficient intersection broadcast mechanism. L. Wischof et al. [8] presented traffic information services. Road conditions and information on traffic jams are disseminated to vehicles and used for navigation and early warning.

Some researchers proposed a bio-inspired metadata harvesting algorithm [9]. The main goal of this algorithm is to effectively cover large urban areas and favor agent movements towards information patches where the concentration of metadata is higher. Zhao et al. [10] proposed a carry-and-forward mechanism for sparsely connected network. It can disseminate vehicular traffic messages when there is no other node between source and destination vehicles. S. Lee et al. [11] proposed a hybrid communication protocol due to the need of exploiting network infrastructure to avoid disconnections in sparse-traffic scenarios. Moreover, the hybrid approach is particularly suitable for comforting applications that use network infrastructures for advertisement dissemination. A prioritized handling of events can be achieved, for instance, by a priority queue as it has been applied in [12]. The authors set different priorities for different events, and then the emergency events have the highest priority than other events. The feasibility is limited due to the excessive control overhead to collect neighbor position including vehicles traveling in the opposite direction.

Greedy Perimeter Stateless Routing (GPSR), a novel routing protocol for vehicle *ad hoc* network that utilizes the positions of routers and a packet's destination to make packet-forwarding decisions [13]. GPSR algorithm uses a greedy algorithm to establish the route. Greedy algorithm is a method that the upstream node chooses a node that is

nearest to the destination from its neighbors as its next hop. When the node finds that its own distance to the destination is the shortest compared with the other distances between its neighbors and the destination node, the node goes into a local minimum problem. Then, the GPSR algorithm goes into perimeter forwarding mode [14]. During this process, the *right-hand* rule combined with face-routing algorithm can solve the Local Minimum Problem. D.Q. Li et al. [15] proposed a new algorithm based on angle and distance (GPSR-AD) for solving the problem which GPSR may produce excessive unwanted route hops when spatial neighbor exists in the *ad hoc* network.

Although the abovementioned data dissemination mechanisms can reduce some network overheads, they do not have an appropriate solution for emergency and personal interested information simultaneously. Based on this, we propose a data dissemination mechanism based on emergency broadcasting and on-demand unicast for inquiry message.

### 3 Broadcasting-based data dissemination

The advantage of our proposed broadcasting scheme is that it abandons the arbitrary method used in the traditional VANETs that might cause a broadcast storm. The arbitrary method causes not only network congestion and subsequent loss of data but also a waste of network resource. Despite some improvements in epidemic protocol and UMB protocol that introduce a probabilistic resilient multicast scheme to avoid broadcast storm, broadcasting messages must be repeated periodically to ensure the message coverage rate in the network. Some improved algorithms such as the topology discovery-based data dissemination algorithms need to transfer the messages and find the next forwarder at the same time. But they need to set the waiting timer for timeout so that the requirements of rapid transferring cannot be satisfied.

In this paper, vehicles get information of their neighbors via hello messages. Before broadcasting, source vehicle adjusts its broadcast radius according to the neighbors' information, and then it chooses some appropriate neighbors to construct a forwarder set. The algorithm can effectively increase message coverage rate and avoid broadcast storm, as well as the network congestion.

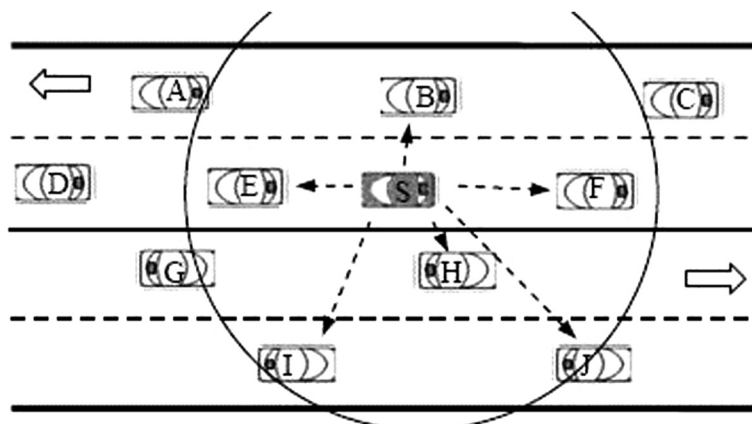
#### 3.1 Assumptions and definitions

In this paper, we have the assumptions as follows:

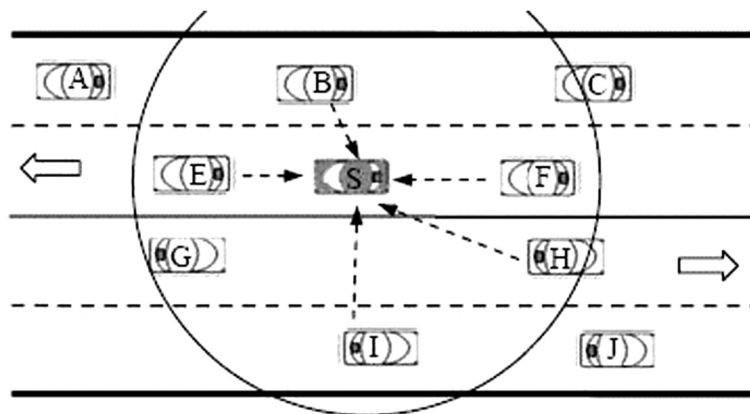
- Each vehicle knows its location through the global positioning system (GPS) service at any given time.
- Each vehicle is equipped with several sensors for monitoring any events and a radio that is capable of short-range wireless communication.
- All vehicles broadcast the hello messages periodically, so that they can know the information (e.g., the current position, speed) of their neighboring vehicles.
- No roadside infrastructure is available during the communication among the vehicles in motorway.
- All vehicles go straight and do not consider the effect of traffic lights during its running in motorway.

The neighbors of a vehicle are usually located in its communication range. We will define the senses by using Figures 1 and 2 as follows.

As is shown in Figure 1, the source vehicle *S* broadcasts a hello message at time  $t_1$ . The hello message will be received by vehicles *B*, *E*, *F*, *H*, *I*, and *J*. In Figure 2, the vehicles that have received the hello messages send a response message to the source vehicle at time  $t_2$ . But the source vehicle *S* will only receive the response messages from vehicles *B*, *E*, *F*, *H*, and *I*, because vehicle *J* moves out of the communication range of vehicle *S*. So, the neighbors of vehicle *S* are vehicles *B*, *E*, *F*, *H*, and *I*, not containing vehicle *J*.



**Figure 1** Source vehicle *S* broadcasts a hello message at time  $t_1$  in the motorway.



**Figure 2** Source vehicle  $S$  receives the response message from its neighbor vehicles at time  $t_2$ .

The hello message includes vehicle's identity, position, velocity, broadcast radius, and time stamp. We define that the interval of broadcasting hello message behaves according to a uniform distribution  $U(0.5P, 1.5P)$ . Here, the symbol  $P$  is an average of historical interval of broadcasting. Therefore, the time a vehicle node receives the hello message from other vehicle nodes is evenly distributed. By this way, each node can update their neighbors' information in time and avoid the broadcast storm as well.

We assume that an emergency event-driven message is generated by an observer vehicle when it passed by the spot where an emergency event occurs (e.g., car accident). The message includes the source vehicle ID, position, timestamp, event type, and forwarders' IDs. The field position stores the geographical coordinates of event, and the field forwarders' IDs indicates all the forwarders, and the emergency event message will be forwarded once it arrived at these nodes.

### 3.2 Prediction for vehicle state

In order to broadcast the emergency message, the source node needs to select part of its neighbors to forward the message. But the state information of neighbors always became outdated, and it may lead to the inaccuracy of the selection of the forwarders. In this section, we propose a weighted  $K$ -nearest neighbors prediction algorithm with self-feedback to predict the current speed of a vehicle. That is to predict the real-time status of a being predicted node by its  $K$ -nearest neighbors' historical data. The innovation of our algorithm is as follows.

1. The method for choosing the number of neighbors' historical data depends on the neighbors' corresponding historical coordinates. We assume that during a range of road, the road condition could be similar and the speed of vehicles could be stable.
2. The value of  $K$  is related to the number of neighbors that the being predicted node can detect. It is

unreasonable to choose a fixed value of  $K$  because of the dynamic neighbors on different roads at different times. So, the principle taken in our algorithm is so long as the distance between neighbors' historical data and the being predicted node's historical data fluctuates within a certain scope, the corresponding neighbor can be regarded as a close neighbor.

3. Instead of using Euclidean distance or Mahalanobis distance, we adopt a new method that uses degree of dispersion to evaluate the distance between historical data, regardless of the dimension of historical velocity vectors.
4. A weighted prediction algorithm is adopted according to the distance between neighbors' historical data and the predicted node's historical data. A neighbor that is closer geographically to the predicted node would have a greater weight.
5. After prediction, the source node evaluates errors between the true value and the prediction value, so that it adaptively adjusts the judgment for being a close neighbor.

Assume that the information of neighbor node  $A$  is detected outdated by source node  $S$  at time  $t$ , then node  $S$  will go the steps of prediction as follows.

1. Extract the broadcast radius of node  $A$  from the latest hello message, called  $R_A$ . Then, traverse the neighbor set of  $S$  (include  $S$ ). For each node  $X$  with timely information in the neighbor set, if the distance between  $A$  and  $X$  is less than  $R_A$ , and the velocity direction of  $X$  is the same as that of  $A$ , then regard  $X$  as a temporary neighbor of  $A$  and put it into a set called  $M$ .
2. For each node  $X$  in set  $M$ , node  $S$  extracts the historical velocity of  $X$  from its hello messages and maintains the historical records for a long term. It is considered in this paper that during a 100-m long section of road, the road condition could be

similar, and the speed of vehicles could be stable. So, we only focus on the historical velocity in a hello message that indicates the node's historical coordinate is within 100 m from the node's current location, and we call them valid historical velocity. Thus, the number of historical velocities extracted can be determined. Suppose there are  $i$  historical velocities  $S$  gets from node  $X$ 's hello messages, then the historical velocity vector of  $X$  is  $V_{Xh} = (v_{X1}, v_{X2}, \dots, v_{Xi})$ . Accordingly, the historical velocity vector of node  $A$  is  $V_{Ah} = (v_{A1}, v_{A2}, \dots, v_{Ai})$ . If  $S$  cannot get enough historical hello messages of node  $A$ , then the dimension of the two velocity vectors is  $\min(i_X, i_A)$ , where  $i_X$  is the number of valid historical velocity of node  $X$ , and  $i_A$  is the number of historical velocity of node  $A$  that  $S$  can get. Notice that, for different nodes in set  $M$ , the dimension of historical velocity vector is different.

- Because historical velocity vectors have different dimensions, Euclidean distance between vectors that have high dimensions would be greater than that having low dimensions. In order to evaluate the distance between neighbors' historical velocity and node  $A$ 's historical velocity, we introduce degree of dispersion. A high discrete degree means a greater distance to node  $A$ . We can calculate the distance between node  $X$  and  $A$  by the following formula.

$$\text{dis}(X, A) = \sqrt{\frac{1}{i_X} * \sum_{j=1}^{i_X} (v_{Xj} - v_{Aj})^2} \quad (1)$$

where  $i_X$  indicates the dimension of historical velocity vectors constructed in step (2). The standard of being a close neighbor to node  $A$  is  $\text{dis}(X, A) \leq r * V_A$ , where  $V_A$  is the velocity of node  $A$  in the latest hello message that node  $S$  received, and  $r$  is a threshold factor set to 0.1 in the first round prediction.

- All the nodes in set  $M$  that satisfy  $\text{dis}(X, A) \leq r * V_A$  will be regarded as close neighbors to node  $A$ . Assume that there are  $K$  close neighbors, then the weight assigned to a close neighbor is as follows.

$$W_X = \frac{\frac{1}{u_X}}{u_2} \quad (2)$$

$$u_X^2 = \text{dis}^2(X, A) = \frac{1}{i_X} * \sum_{j=1}^{i_X} (v_{Xj} - v_{Aj})^2 \quad (3)$$

$$u^2 = \sum_{X=1}^K \frac{1}{u_X^2} \quad (4)$$

where  $u_X^2$  reflects the distance between node  $X$  and node  $A$ . Evidently though, a higher weight will be assigned to a node that is closer to node  $A$ .

- The final prediction for the real-time velocity of node  $A$  is:

$$V\tilde{A} = \sum_{X=1}^K \left( W_X * \frac{1}{i_X} * \sum_{j=1}^{i_X} v_{Xj} \right) \quad (5)$$

- Define a variation factor  $\delta = V_{\tilde{A}}/V_A$ , where  $V_A$  represents the velocity of  $A$  in the latest hello message that  $S$  has received.

- Predict the distance between  $A$  and  $S$  at time  $t$ .

$$d\tilde{SA} = \sqrt{(x\tilde{A} - x_S)^2 + (y\tilde{A} - y_S)^2} \quad (6)$$

where  $x_S$  and  $y_S$  represent the coordinates of node  $S$  at the calculation time.  $x_{\tilde{A}}$  and  $y_{\tilde{A}}$  are the prediction coordinates of node  $A$  and can be calculated as follows.

$$\frac{x\tilde{A} - x_A + V\tilde{A} - V_A}{2(t - t_A)} \quad (7)$$

$$y\tilde{A} = y_A \quad (8)$$

The parameter  $t_A$  represents the timestamp in the latest hello message of node  $A$  that  $S$  has received.  $x_A$  and  $y_A$  are the coordinates of node  $A$  at time  $t_A$ . We consider that node  $A$  and node  $S$  move in the same direction or in the opposite direction, so the  $y$  coordinate remains unchanged.

- The prediction broadcast radius of node  $A$  at time  $t$  can be calculated as  $R_{\tilde{A}} = R_A * \delta$ , where  $R_A$  is the broadcast radius of  $A$  in the latest hello message that  $S$  has received. In the worst case, there is no eligible node in step (1), and then the original values of  $d\tilde{SA}$  and  $R_{\tilde{A}}$  will be retained.
- Before the next prediction, node  $S$  can get node  $A$ 's last hello message. Extract the velocity  $V'_A$  and regard it as a true value at the prediction time. Define the relative error  $E = |V'_A - V\tilde{A}|/V'_A$  and adaptively adjusts the threshold factor  $r$  to reduce the relative error.

### 3.3 Construct the set of forwarders

In order to reduce the influence of broadcast storm, only a part of appropriate neighbors will be selected to forward the emergency message. The set of forwarders of  $S$  can be constructed as follows.



1. Define the set of forwarders of  $S$  called  $F_S$ , and then put the neighbor node of  $S$  which has the maximum distance to  $S$  into  $F_S$ .
2. Define the rest neighbors of  $S$  as a sequence  $\{X_i \mid i = 1, 2, \dots, n-1\}$ , where  $n$  is the number of neighbors of  $S$ , and  $X_i$  is sorted by the distance between  $X_i$  and  $S$  from the greatest to smallest. The nodes in  $F_S$  is called  $Y_j$  ( $j = 1, 2, \dots, m$ , where  $m$  is the number of nodes in  $F_S$ ). For a given  $X_i$ , if the distance between  $X_i$  and  $Y_j$  is greater than the latest broadcast radius  $R_{Y_j}$  for all  $Y_j$  ( $j = 1, 2, \dots, m$ ), then put  $X_i$  into  $F_S$ . Otherwise, abandon  $X_i$ .
3. Traverse all neighbors of  $S$ , and eventually,  $F_S$  contains all forwarders of  $S$ .

### 3.4 The algorithm for broadcasting emergency message

The source vehicle  $S$  receives hello messages from its neighbors at a random time and save the information into its neighbor list. If there is a new hello message from node  $A$ , then the information of  $A$  in the neighbor list will be updated. Since node  $S$  cannot obtain the future status of its neighbors, it will execute the prediction algorithm to make its neighbors' information more accurate.

Once an emergency event is detected by node  $S$  at time  $t$ , then the broadcasting for emergency messages starts as follows.

1. The source vehicle  $S$  traverses the neighbor list and extracts the information of its neighbors.
2. For each neighbor of  $S$ , if the difference between its timestamp (the timestamp field in its latest hello message) and current time is less than or equal to an overdue threshold  $dl$ , then the information of this neighbor node is considered as real time. Otherwise,  $S$  executes the prediction algorithm for this neighbor node. The threshold  $dl$  is recommended to be set to  $P/2$ , where  $P$  is an average of historical interval of broadcasting.
3. If all the neighbors' information is real time or has been updated by the predicted value, then  $S$  calculates its broadcast radius and constructs the set of forwarders  $F_S$ .
4.  $S$  constructs the emergency event message and fills the last field with the forwarder IDs in set  $F_S$ , and then it broadcasts this message.

The forwarders also need to broadcast the emergency event messages of source node  $S$ , so that each forwarder will execute the broadcasting algorithm mentioned above. Then, the emergency event message can spread rapidly in the network.

In the proposed broadcasting algorithm, the broadcasting radius can be adjusted according to its surroundings. By this way, on the one hand, the situation that some

nodes cannot be covered because of a small broadcast radius can be avoided. On the other hand, the problem of signal attenuation caused by a too large broadcast radius can also be solved. Therefore, the proposed broadcasting algorithm improves the coverage rate of nodes in the network. We construct the forwarders set  $F_S$ , and the overall network can be covered using less nodes. The introduction of overdue threshold  $dl$  ensures that the information saved in neighbor lists can be closer to a true value. By analyzing the latest status of neighbors, we hope that the prediction can be more accurate.

## 4 The on-demand unicast algorithm for inquiry message

While as to simple user, he/she may have end-to-end communication demands. As all of vehicles need not respond the request messages from a specific individual, we design an on-demand unicast scheme for these messages. The fields of information request message must include the type of information (e.g., parking place, gas station location, traffic congestion information, etc.) and deadline of receiving reply. The message should be broadcasted in order to acquire timely reply while the reply is sent by unicast.

### 4.1 Reply sorting

When a source node has some reply messages to transmit, it should sort the messages to satisfy deadlines of different requestors. As a node only transmits a message at one time, the source selects a node according to the expected deadline. It is assumed that  $E_{ij}(t)$  is the emergency level for reply message  $j$  to node  $i$  at time  $t$ .

$$E_{ij}(t) = \frac{T_{ij} - t - MT_{ij}}{\frac{1}{n} * \sum_{k=1}^n (T_{kj} - t)} \quad (9)$$

$\{P_1, P_2, \dots, P_n\}$  is the set of destination nodes (requestors). For each node  $P_i$ ,  $T_{ij}$  is the deadline for transmitting message  $j$  to node  $P_i$  ( $i = 1, 2, \dots, n$ ).  $MT_{ij}$  is the transmission time for message  $j$  to node  $P_i$ . The source evaluates the transmission time for each requestor. The more the value  $E_{ij}(t)$  is, the sooner the message  $j$  is sent. The overhead of this sorting process is related to the number of messages the source node needs to reply, and the time computational complexity of formula (9) is  $O(n)$ , where  $n$  is the number of destination nodes. Through ranking the sequence for messages, the source vehicle can decide which message should be sent at first.

### 4.2 Data dissemination route protocol

In the paper, the algorithm is concerned to data distribution in the motorway. In the original GPSR algorithm, the data distribution process can be described as follows:

1. Greedy mode. An upstream node selects some of its neighbor nodes which is nearest to the destination and regards it as the next hop. When the node finds that its own distance to the destination is the shortest path compared with the distance between each of neighbors and the destination node, then the algorithm turns into perimeter forwarding mode.
2. Perimeter forwarding mode. During this process, the *right-hand* rule combined with face-routing algorithm is applied to find the next hop. Though the *right-hand* rule can ensure the upstream node to find its next hop, the choice of the next hop is accompanied by arbitrariness, which may lead to more route hops.

In order to reduce computation complexity and avoid excessive unwanted route hops caused by *right-hand* rule, we maintain the greedy mode and modify the perimeter forwarding mode to select an appropriate next hop node whose distance and angle are near the destination. When it turns into perimeter forwarding mode, then the node will calculate the next hop neighbor. We will find the neighbor of  $N_i$  with minimum  $D_{i+1}$ .

$$D_{i+1} = \frac{\text{dis}(N_{i+1}, G)}{\text{dis}(N_i, G)} + \frac{\angle N_{i+1}N_iG}{\angle XN_iG} \quad (10)$$

In Figure 3a, node  $G$  is the destination, and  $N_i$  is the current forwarding node. The expressions  $\text{dis}(N_{i+1}, G)$  and  $\text{dis}(N_i, G)$  respectively depict the distance between  $N_{i+1}$  and  $N_i$  to  $G$ . The node  $X$  is an intersection of two circles whose centers are respectively  $N_i$  and  $G$ . According to the formula (10), we find the neighbor of  $N_i$  with minimum  $D_{i+1}$ . Then, the corresponding node  $N_{i+1}$  is the next hop node. By this algorithm, when greedy mode fails, the current forwarding node chooses a next hop node which is closer to the destination in the respects of distance and angle [14]. So, the multi-hop and fallback problems in GPSR can be solved.

In Figure 3b, we can observe the significant advantage of GPSR-AD clearly. The arrows mean the information stream direction. Node  $D$  is the destination, and when the message reaches node  $C$ , there is no other node closer to node  $D$  than node  $C$ , so the algorithm turns into perimeter forwarding mode. The forward routing path is  $C-K-E-F-G-H-I-J-L-D$  according to *right-hand* rule in the original GPSR. But by formula (10), the forward routing path can be simplified to  $C-H-I-J-L-D$ . It is obvious that GPSR-AD algorithm reduces route hops and improves the efficiency.

#### 4.3 Demand-driven data dissemination mechanism

The data distribution is designed as follows:

1. If a source node  $S$  receives more than one inquiry message, then  $S$  should calculate emergency level for

each message and determine the order of message delivery. According to the locations of all the requestors,  $S$  classifies them into *forward* and *backward* groups. The former consists of the requestors locate in front of the source, and the latter includes nodes at the back (Figure 4 is an example of the location group). The angle  $\theta$  is the angle between the positive  $Y$ -axis and the line from  $S$  to any requestors. If  $\sin\theta \geq 0$ , then the requestor is a *forward* node, or else is a *backward* node. In Figure 4, node  $E$ ,  $F$ , and  $G$  are *forward* nodes, and node  $A$ ,  $B$ ,  $C$ , and  $D$  are *backward* nodes.

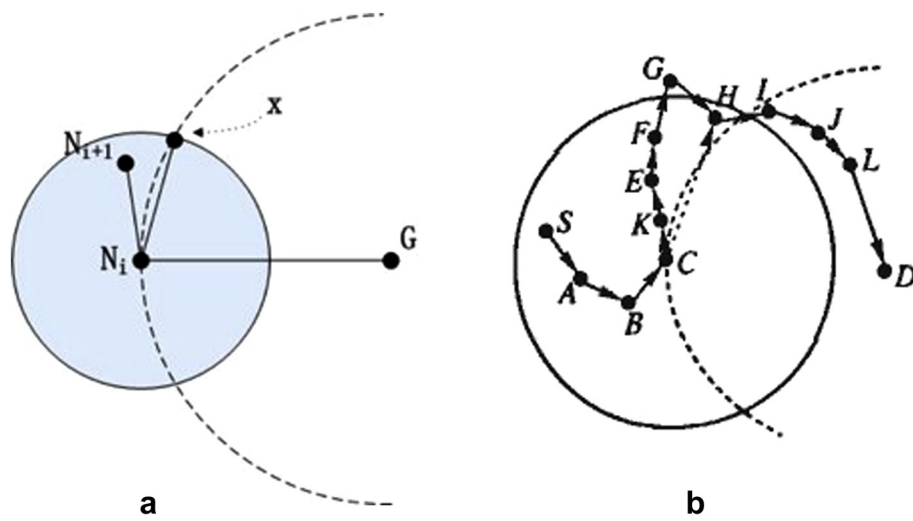
2. Node  $S$  forms two destination address lists according to *forward* and *backward* groups, and then constructs two reply messages filled with the destination address lists and message types (specifically, parking place, gas station location, traffic congestion information, etc.). When  $S$  starts to transmit the reply messages, it selects the nearest requestor (e.g.,  $A_1$ ) from the address list as the destination and transmits the packet using improved GPSR algorithm.
3. When the reply message arrives to  $A_1$ , node  $A_1$  acquires the message content, deletes itself from the list, and selects a nearest requestor as next destination to forward the message.
4. Any forwarding node should calculate the urgent level to determine the order of message delivery if it needs to forward more than one inquiry message. If any request exists in the destination list, the algorithm returns to step (3), or else this type of message is sent completed.
5. If a requestor receives the reply before the deadline, then the procedure of *request-response* is successful, or else it repeats the request in view of its demands.

## 5 Performance evaluation

The number of vehicles in actual world is various, and the road structure is complex, so we choose a type of motorway model as shown in Figure 4. All vehicles go straight and cannot change driveway during its running. The vehicle velocity is randomly set but still limited by traffic density. In addition, the traffic lights are not considered in motorway. In our simulation, the number of vehicles ranges from 100 to 700 and the velocity from 60 to 100 km/h. The MAC protocol is 2 Mbps 802.11p. The proposed algorithms are coded in MATLAB.

### 5.1 Simulation of broadcasting of emergency message

Our algorithm (DSV-BD) is compared with the UMB protocol and the epidemic scheme. In these two algorithms, broadcasting radius is set to 50 m. The simulation is repeated for 30 times, and the results are the average of those experiments. There are three indicators in our simulations:



**Figure 3** Illustration of a formula and comparison between GPSR and GPSR-AD. (a) The illustration for formula (10); (b) Comparison between GPSR and GPSR-AD.

*Delivery ratio*: a probability defined as the number of message receivers divided by the total number of vehicles in the networks.

*Number of forwarders*: the number of vehicles which forward the message in a round of broadcasting. Generally, a smaller number of forwarders indicate less delay of message and lower network loads.

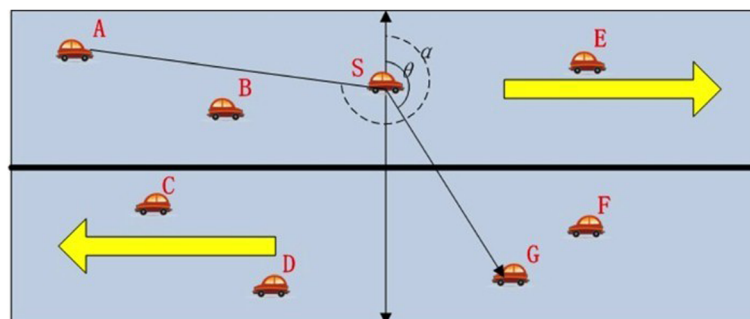
*Number of message copies*: the number of message copies in a round of broadcasting. Once a forwarder receives a message, it generates a copy. This indicator reflects the network congestion level and influences the delivery ratio and transmission delays.

It is known that the traffic density is varying by different roadsides and different times. So, we study the performance of the algorithm for various traffic densities by varying the total number of vehicles in the network. The size of this simulation scenario is  $1,000 \text{ m} \times 50 \text{ m}$ . We assume that the number of vehicles increases from 100 to 700.

Also, message coverage area has an influence on three abovementioned indicators. It reflects the distribution and spreading ability of emergency messages. We adjust the length of road from 500 to 3,500 m to simulate the change of message coverage area. In this simulation scenario, the number of vehicles is set to 500 and remains unchanged.

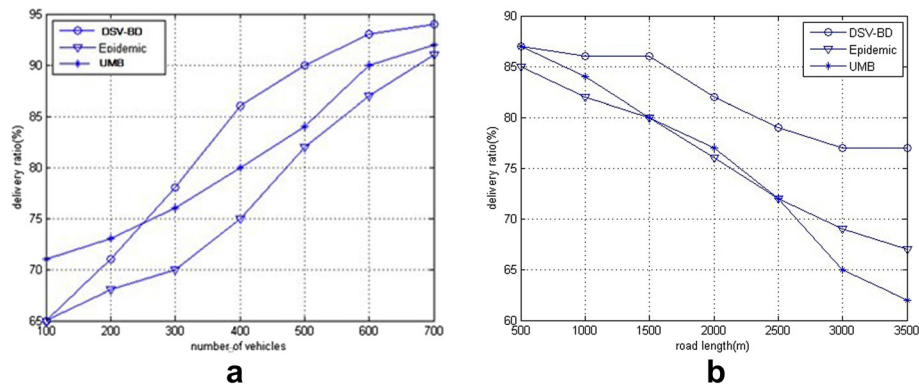
#### 5.1.1 Delivery ratio

Figure 5a shows the comparison of data delivery ratio in the three algorithms in the case of different number of vehicles. When the number of vehicles is lower, delivery ratio is about 70.8% in these algorithms. But as the number of vehicles increases, delivery ratio increases linearly over 90.3%. This is because as the number of vehicles increases, the number of forwarders increases accordingly, so that the emergency message can cover more vehicle nodes. However, the delivery ratio of DSV-BD is higher



**Figure 4** An example of location group.





**Figure 5** Comparison of data delivery ratio with different number of vehicles and different message coverage areas. (a) Data delivery ratio with different number of vehicles. (b) Data delivery ratio with different message coverages.

than other protocols especially when the number of vehicles is higher. The average delivery ratio of DSV-BD is 6.92% higher than epidemic and 1.95% higher than UMB when the number of vehicles is 700.

Figure 5b shows the comparison of data delivery ratio in the three algorithms in the case of different message coverage area. From the chart, we can see that data delivery ratio has a sharp decrease due to data collisions. This is because as the length of road increases, the message coverage area increases accordingly. So, that delay of messages increases and leads to message dropping. By comparison, the delivery ratio of DSV-BD has the slowest descending speed and is 7.90% higher than that of epidemic and 8.91% higher than that of UMB on average.

### 5.1.2 Number of forwarders

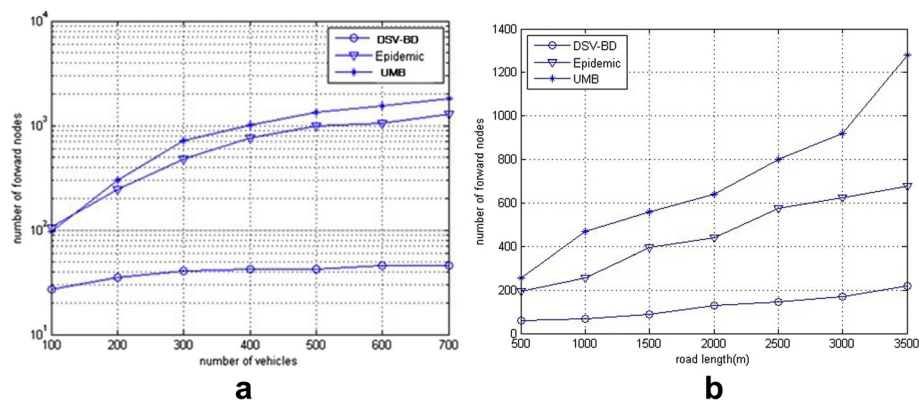
Figure 6a shows that in case of the different number of vehicles, the number of forwarders in broadcasting varies. It increases linearly as the number of vehicles increases. Also, it is clearly observed that DSV-BD only needs fewer forwarders compared with other two algorithms. The fitting

calculations for the three slopes of the line are DSV-BD 0.0317, epidemic 2.156, and UMB 2.86. The number of forwarders in our algorithm rises slowly and has a distinct advantage.

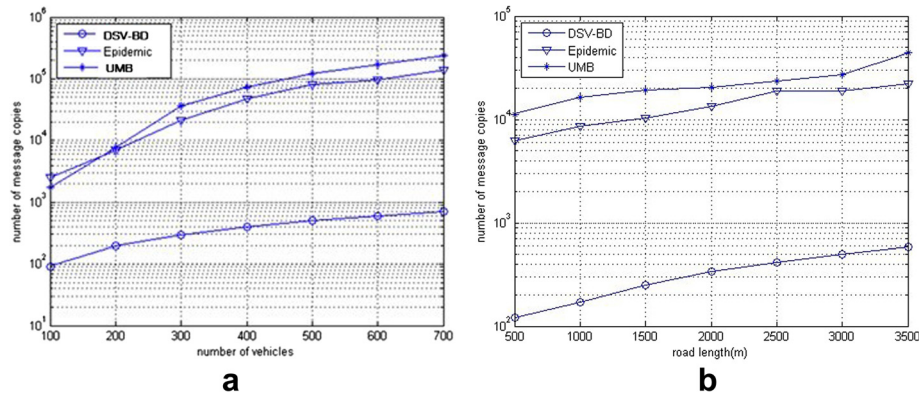
In Figure 6b, with increasing length of road, the number of forwarders increases linearly. The fitting calculations for the three slopes of the line are DSV-BD 0.143, epidemic 0.488, and UMB 0.854. It is obvious that our algorithm has fewer forwarders than the other two algorithms, so the transmission time and network loads can be reduced.

### 5.1.3 Number of message copies

Figure 7a shows the comparison of number of message copies in the three algorithms in the case of the different number of vehicles. The number of message copies reflects the congestion status of networks, and the network congestion influences the transmission delay directly. In DSV-BD algorithm, less forwarders lead to less message copies. As seen from the chart, with increasing of the



**Figure 6** Comparison of different number of forwarders and different message coverage areas. (a) Number of forwarders with different number of vehicles. (b) Number of forwarders in different message coverages.



**Figure 7** Comparison of different number of message copies and different message coverage areas. (a) Number of message copies with different number of vehicles. (b) Number of message copies with different message coverages.

number of vehicles, number of message copies of epidemic and UMB methods increase obviously.

In Figure 7b, the number of message copies increases as the length of road increases. By comparison, the growth rates are rapidly increasing in epidemic and UMB, but slowly in our algorithm. The number changes from 2.2% to 5.1%. So our method can reduce the network load effectively.

## 5.2 Simulation of on-demand unicast algorithm

The two compared algorithms are involved:

- Greedy: the upstream forwarders always choose the requestor which has the shortest distance to them, and the reply messages will be sent according to this order.
- Deadline-pri: the upstream forwarders always choose the requestor which has a smaller deadline, and the reply messages will be sent according to this order.

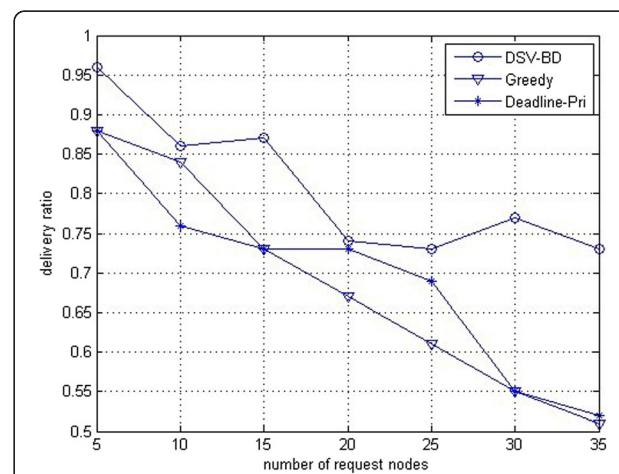
The indicator involved is delivery ratio, which means the probability defined as the number of message receivers divided by the total number of vehicles in the networks. We aim at the impact of the number of requestors. The deadline of each requestor obeys uniform distribution from 0 to 6 s. The simulation is repeated for 30 times, and the results are the average of those experiments. Figure 8 shows the comparison of data delivery ratio in these three algorithms in the case of different number of requestors.

As the number of requestors increases, the delivery ratios of three algorithms decrease generally. In terms of the two compared algorithms, the more the requestors, the higher the queuing time is. This leads to higher probability of timeout. As to DSV-BD, the increase of requestors means the increase of hops in the delivery path and more timeout happens due to increasing delivery time. As is shown from the figure above, the delivery ratio of DSV-BD is 17.3%

higher than that of greedy and 15.9% higher than that of deadline-pri, which shows the higher delivery reliability with various numbers of requestors.

## 6 Conclusions

The high-speed moving of the vehicles in motorway environment results in the highly dynamic topology of networks, which brings out challenges for data dissemination of VANETs. In this paper, we propose a new broadcasting algorithm and an on-demand unicast algorithm for VANETs. In the proposed broadcasting algorithm, there are two innovation points: 1) predict the status of nodes by an improved  $K$ -nearest neighbors' prediction and 2) construct the set of forwarders to alleviate the influence of broadcast storm. In the on-demand unicast algorithm, the source vehicle node only sends at most two messages, and all the requestors can receive the messages they need. The results show that our method can achieve better performance with higher delivery ratio and less number of message copies.



**Figure 8** Data delivery ratio in different number of requestors.

**Competing interests**

The authors declare that they have no competing interests.

**Acknowledgements**

This work was partly supported by the NSFC (61372108), Ph.D. Programs Foundation of Ministry of Education of China (No. 20110005110011), Fundamental Research Funds for the Central Universities (No. 2014RC1102), and Beijing Higher Education Young Elite Teacher Project (YETP0474).

**Author details**

<sup>1</sup>State Key Laboratory of Networking and Switching Technology, Beijing University of Posts and Telecommunications, No. 10 Xi Tu Cheng Road, Beijing 100876, China. <sup>2</sup>China Electronic Systems Engineering Corporation, Xiaotun Road, Beijing 100039, China.

Received: 5 August 2014 Accepted: 2 March 2015

Published online: 02 April 2015

**References**

1. K Rakesh, D Mayank, A Framework for Handling Local Broadcast Storm Using Probabilistic Data Aggregation in VANET. *Wireless Personal Communication* **72**, 315–341 (2013)
2. W Shangguang, F Cunqun, H Ching-Hsien, S Qibo, Y Fangchun, A Vertical Handoff Method via Self-selection Decision Tree for Internet of Vehicles, in *IEEE System Journal*, 2014. doi: 10.1109/JSYST.2014.2306210
3. S Michael, M Imad, Spatial distribution and channel quality adaptive protocol for multihop wireless broadcast routing in VANET. *IEEE Trans Mobile Comput* **12**(4), 722–734 (2013)
4. SY Ni, YC Tseng, YS Chen, JP Sheu, The Broadcast Storm Problem in a Mobile Ad Hoc Network. *Wireless Networks* **8**, 153–167 (2002)
5. S Panichpapiboon, W Pattara-atikom, A Review of Information Dissemination Protocols for Vehicular Ad Hoc Networks. *Communications Surveys & Tutorials IEEE* **99**, 1–15 (2011)
6. Korkmaz, G., Ekici, E., Ozguner, F., and Ozguner, U.: Urban Multi-hop Broadcast Protocol for Inter-vehicle Communication Systems. In: *Proc. 1st ACM International Workshop on Vehicular Ad Hoc Networks. (VANET'04)*, Philadelphia, USA, pp. 76–85 (2004)
7. Korkmaz, G., Ekici, E., and Ozguner, F.: An Efficient Fully Ad-hoc Multi-hop Broadcast Protocol for Inter-vehicular Communication Systems. In: *Proc. IEEE International Conference on Communications. (ICC'06)*, Istanbul, Turkey, pp. 423–428 (2006)
8. L Wischof, A Ebner, H Rohling, Information Dissemination in Self Organizing Intervehicle Networks. *Intelligent Transportation Systems, IEEE Transactions on* **6**, 90–101 (2005)
9. L Uichin, M Eugenio, G Mario, B Paolo, L Pietro, L Kang-Won, Bio-inspired multi-agent data harvesting in a proactive urban monitoring environment. *Ad Hoc Networks* **7**(4), 725–741 (2009)
10. J Zhao, G Cao, VADD: Vehicle-Assisted Data Delivery in Vehicular Ad Hoc Networks. *Proceedings of IEEE International Conference on Computer Communications (INFOCOM'06)*, 2006, pp. 1–12
11. Lee, S., Pan, G., Park, J., Gerla, M., and Lu, S.: Secure Incentives for Commercial Ad Dissemination in Vehicular Networks. In: *ACM international symposium on Mobile ad hoc networking and computing (MobiHoc)*, New York, NY, USA, pp. 150–159 (2007)
12. M Röckl, P Robertson, Data Dissemination in Cooperative ITS from an Information-Centric Perspective. *International Conference on Communications (ICC)*, Cape Town, South Africa, pp. 1–6 (2010)
13. Karp, B., and Kung, H.T.: GPSR: Greedy Perimeter Stateless Routing for Wireless Networks. In: *Proceedings of the Annual International Conference on Mobile Computing and Networking (MobiCom'00)*, Boston, USA, pp. 243–254 (2000)
14. Hill, J., Szewczyk, R., Woo, A., Hollar, S., Culler, D., and Pister, K.: System Architecture Directions for Networked Sensors. In: *International Conference on Architectural Support for Programming Languages and Operating Systems (ASPLOS)*, New York, NY, USA, pp. 93–104 (2000)
15. L Daoquan, L Haiyan, C Qiguang, W Huaicai, New routing algorithm based on geographical location: GPSR-AD. *J Comput Appl* **29**(12), 3215–3217 (2009)

**Submit your manuscript to a SpringerOpen<sup>®</sup> journal and benefit from:**

- Convenient online submission
- Rigorous peer review
- Immediate publication on acceptance
- Open access: articles freely available online
- High visibility within the field
- Retaining the copyright to your article

---

Submit your next manuscript at ► [springeropen.com](http://springeropen.com)